

Docket No.: P2001,0023

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to: Commissioner for Patents, Alexandria, VA 22313 20231.

By: 

Date: September 8, 2003

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant : Heimo Hartlieb et al.  
Appl. No. : 10/621,536  
Filed : July 17, 2003  
Title : Method for Increasing the Security of a CPU

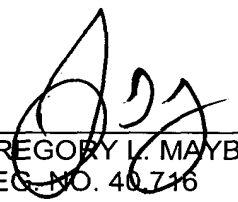
CLAIM FOR PRIORITY

Hon. Commissioner for Patents,  
Alexandria, VA 22313-1450  
Sir:

Claim is hereby made for a right of priority under Title 35, U.S. Code, Section 119, based upon the German Patent Application 101 01 956.4 filed January 17, 2001.

A certified copy of the above-mentioned foreign patent application is being submitted herewith.

Respectfully submitted,

  
\_\_\_\_\_  
GREGORY L. MAYBACK  
REG. NO. 40,716

Date: September 8, 2003

Lerner and Greenberg, P.A.  
Post Office Box 2480  
Hollywood, FL 33022-2480  
Tel: (954) 925-1100  
Fax: (954) 925-1101

/mjb

# BUNDESREPUBLIK DEUTSCHLAND



## Prioritätsbescheinigung über die Einreichung einer Patentanmeldung

**Aktenzeichen:** 101 01 956.4  
**Anmeldetag:** 17. Januar 2001  
**Anmelder/Inhaber:** Infineon Technologies AG,  
München/DE  
**Bezeichnung:** Verfahren zur Erhöhung der Sicherheit  
einer CPU  
**IPC:** G 06 F 12/14

**Die angehefteten Stücke sind eine richtige und genaue Wiedergabe der ursprünglichen Unterlagen dieser Patentanmeldung.**

München, den 17. Juli 2003  
**Deutsches Patent- und Markenamt**  
**Der Präsident**  
Im Auftrag

Agurks

## Beschreibung

### Verfahren zur Erhöhung der Sicherheit einer CPU

- 5 Die vorliegende Erfindung betrifft ein Verfahren zur Verbesserung der Sicherheit einer CPU.

Differential Power Analysis (DPA) ist ein bekanntes Angriffsszenario für Sicherheits-CPU's. Bei einem solchen Angriff wird  
10 eine Folge von Programmbefehlen und deren Auswirkungen in der CPU mittels statistischer Auswertungen der Kennlinien des Stromverbrauchs ermittelt. Aus diesen Auswertungen lassen sich detaillierte Rückschlüsse über das ausgeführte Programm gewinnen.

15 Aufgabe der vorliegenden Erfindung ist es, ein Verfahren zur Erhöhung der Sicherheit einer CPU anzugeben.

Diese Aufgabe wird mit dem Verfahren mit den Merkmalen des  
20 Anspruches 1 gelöst. Ausgestaltungen ergeben sich aus den abhängigen Ansprüchen.

Bei dem erfindungsgemäßen Verfahren wird eine als Pipeline aufgebaute CPU mit mindestens einer Decodierstufe und einer  
25 Rückspeicherstufe verwendet, die typisch eine Ladestufe (Fetch Stage), eine Decodierstufe (Decode Stage), eine Ausführungsstufe (Execute Stage) und eine Rückspeicherstufe (Writeback Stage) umfasst. Die Rückspeicherstufe besitzt mindestens ein Register, bei dessen Benutzung keine Zustandsänderung der CPU erfolgt, und mindestens ein Register, bei dessen  
30 Benutzung eine Zustandsänderung der CPU erfolgt. Erfindungsgemäß wird in der Decodierstufe mindestens eine zufällig ausgewählte Codesequenz als Platzhalter-Code oder Füllsel eingefügt. Dieses Verfahren ist im Prinzip für beliebige  
35 Pipelines anwendbar, die insbesondere zusätzlich zu den als Beispiel angegebenen Stufen über weitere Stufen verfügen können, und wird anhand der beigefügten Figuren näher erläutert.

Die Figur 1 zeigt ein Diagramm der beschriebenen Pipeline. Die Figur 2 zeigt ein Schema für das Vorgehen beim Einfügen der Codesequenzen.

5 In der Figur 1 ist ein Ablaufdiagramm dargestellt, das den Programmablauf von der Ladestufe 1 über die Decodierstufe 2 in die Ausführungsstufe 3 und von dort in die Rückspeicherstufe 4 einer als Beispiel dargestellten Pipeline zeigt. Die Rückspeicherstufe 4 besitzt hier mindestens ein erstes Register 41 als Scratch-Register und ein zweites Register 42 als Writeback-Register. Das Scratch-Register ist ein Register, bei dessen Benutzung keine Zustandsänderung der CPU erfolgt, während bei der Benutzung des Writeback-Registers eine Zustandsänderung der CPU erfolgt. Zur Erhöhung der Sicherheit  
10 der CPU wird von der Decodierstufe 2 eine Codesequenz, und zwar im Prinzip eine beliebige Codesequenz, in den Programmcode, der in der Pipeline übermittelt wird, eingeschleust. Es ist auch möglich, an mehreren Stellen des Programmcodes eine jeweilige zusätzliche Codesequenz als Platzhalter oder Füllsel (dummy code sequence) einzufügen. Das ist in der Figur 2  
15 im Schema dargestellt.

Die Figur 2 zeigt im Schema eine Codesequenz 5 eines beliebigen Programms. In dieser Codesequenz 5 werden zufällig ausgewählte Codesequenzen 6 (Dummy-Sequenzen) an verschiedenen vorgegebenen oder ebenfalls zufällig ausgewählten Stellen eingefügt, so dass sich die erweiterte Codesequenz 50 ergibt. Die eingefügten Codesequenzen können zum Beispiel aus einem Speicher, insbesondere aus einem ROM, ausgelesen werden.  
25

30 Die einzelnen Befehle zum Einfügen von Codesequenz können beispielsweise durch den Abruf von Adressen, die ein Zufallszahlengenerator erzeugt, generiert werden. Die einzufügenden Codesequenzen werden aus dem Speicher ausgelesen und an den Decoder in zufälliger Länge und Reihenfolge übermittelt. Der Decoder schleust den Code dieser Dummy-Codesequenzen in den laufenden Programmcode (Codestream) ein. Auch die Adressen,

an denen der zufällig ausgewählte Code in den Programmcode eingeschleust wird, können mit einer an sich bekannten Zufallsmethode ermittelt werden.

- 5 Durch die zufallsbedingt eingefügte Codesequenz oder die mehreren zufällig ausgewählten und eingefügten Codesequenzen, die nur als Platzhalter oder Füllsel fungieren, wird keine Zustandsänderung der CPU hervorgerufen. Ein wesentlicher Vorteil dieses Verfahrens ist dabei, dass sich die Ausführungs-
- 10 zeit des eigentlichen Programmcodes bei jedem Durchlauf desselben Programms gegenüber den vorhergehenden Durchläufen beliebig verändern lässt und dadurch ein Angriffsversuch, welchem statistische Auswertungen zugrunde liegen (wie zum Beispiel der eingangs erwähnten DPA), wesentlich erschwert ist.

Patentansprüche

1. Verfahren zur Erhöhung der Sicherheit einer CPU, bei dem eine Pipeline aus mindestens einer Decodierstufe (2) und einer Rückspeicherstufe (4) mit mindestens einem ersten Register (41), bei dessen Benutzung keine Zustandsänderung der CPU erfolgt, und mit mindestens einem zweiten Register (42), bei dessen Benutzung eine Zustandsänderung der CPU erfolgt, eingesetzt wird,  
5  
10 d a d u r c h g e k e n n z e i c h n e t, dass in der Decodierstufe (2) mindestens eine zufällig ausgewählte Codesequenz als Platzhalter-Code oder Füllsel eingefügt wird, die keine Zustandsänderung der CPU bewirkt.
- 15 2. Verfahren nach Anspruch 1, bei dem eine oder mehrere zufällig ausgewählte Codesequenzen aus einem Speicher anhand einer bzw. mehrerer zufällig ermittelter Speicheradressen ausgelesen werden.
- 20 3. Verfahren nach Anspruch 2, bei dem als Speicher ein ROM verwendet wird.
4. Verfahren nach einem der Ansprüche 1 bis 3, bei dem zusätzliche Mittel vorhanden sind, die dafür vorgesehen sind, sicherzustellen, dass bei jedem Durchlauf eines bestimmten  
5 Programms eine als Platzhalter-Code oder Füllsel verwendete und zufällig ausgewählte Codesequenz derart ausgewählt wird, dass eine jeweils von vorhergehenden Programmdurchläufen verschiedene Ausführungsdauer des Programms bewirkt wird.

## Zusammenfassung

### Verfahren zur Erhöhung der Sicherheit einer CPU

5 Bei dem Verfahren wird eine Pipeline bestehend aus einer La-  
destufe (1), einer Decodierstufe (2), einer Ausführungsstufe  
(3) und einer Rückspeicherstufe (4) verwendet. Die Rückspei-  
cherstufe besitzt mindestens ein Register (41), bei dessen  
Benutzung keine Zustandsänderung der CPU erfolgt, und mindes-  
10 tens ein Register (42), bei dessen Benutzung eine Zustands-  
änderung der CPU erfolgt. Erfindungsgemäß wird in der Deco-  
dierstufe mindestens eine zufällig ausgewählte Codesequenz  
als Platzhalter-Code oder Füllsel eingefügt, womit ein An-  
griff durch DPA erschwert wird.

15

Figur 1

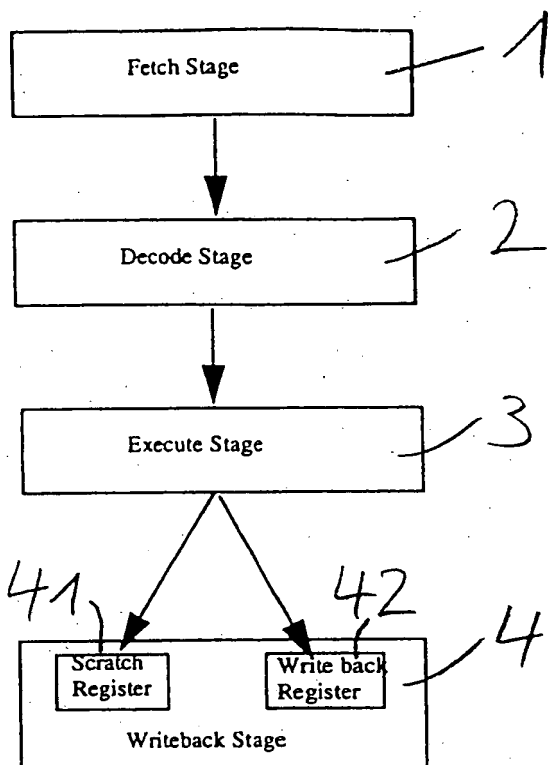


Fig 1

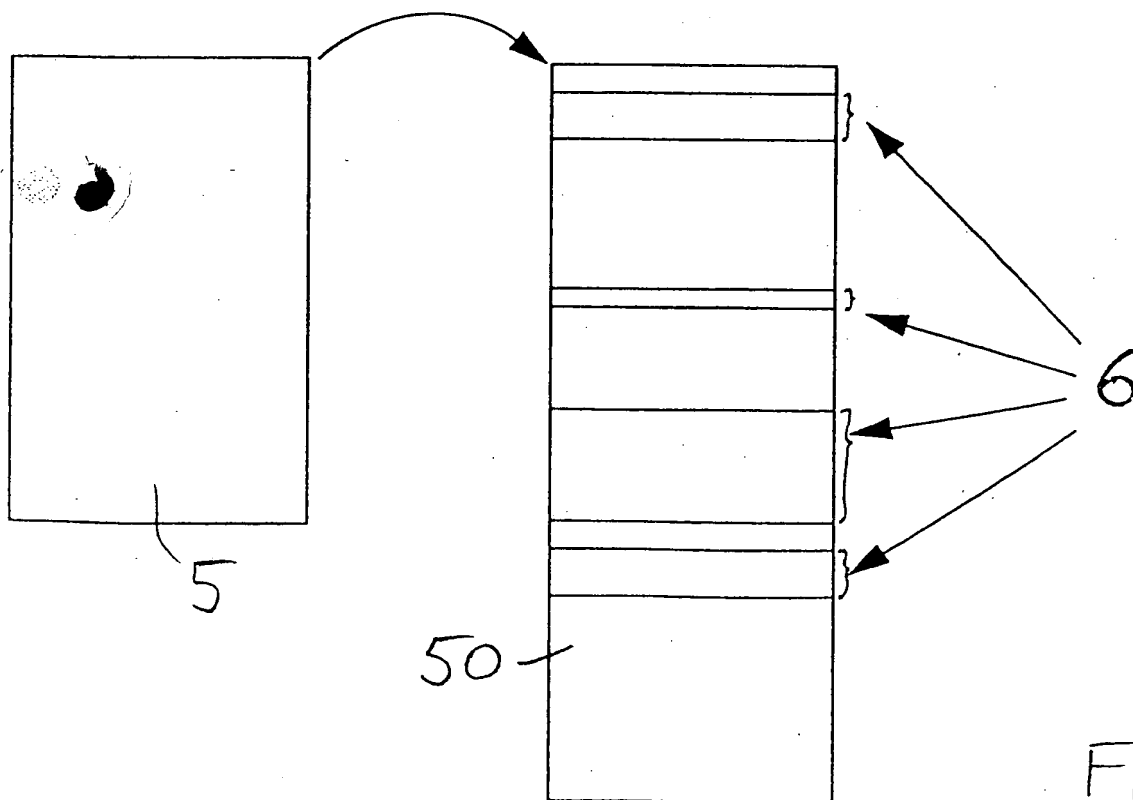


Fig 2